OptimoDE: Programmable Accelerator Engines Through Retargetable Customization

Nathan Clark, Hongtao Zhong, Kevin Fan, Scott Mahlke
CCCP Research Group
University of Michigan
http://cccp.eecs.umich.edu

Krisztián Flautner, Koen Van Nieuwenhove
ARM Limited

OptimoDE Overview

• OptimoDE
  – A configurable VLIW-styled Data Engine architecture
  – Targeted at intensive data processing

• Characteristics
  – Very wide performance envelope
    • Power / area / speed tradeoff
    • Exploiting parallelism in applications
  – Unlimited data path configuration options
  – User extensible through ISA customization

• Semi-automatic design system
  – User-in-the-loop design, retargetable compiler toolchain
OptimoDE in a System On Chip

SDRAM
SDRAM Controller
ARM
CPU
SRAM
I/O
AHB Bus Matrix
DMA Controller
Interrupt Controller
Data Engine
CTRL
MEM
FIFO
Switch
Memory
Control
DMA Controller
Interrupt Controller

OptimoDE Architecture Model

Function units
Memories
I/O ports

Controller
Interconnect
Reg
Reg
Reg
Reg
Reg

I/O ports

Functional Units
- ALU, ACU, Multipliers
- Custom
Memory
- RAM (async / synch)
- ROM
I/O ports
- addressable
- handshake protocol
Registers
- Register files
Interconnect
- Direct connection
- Shared bus
Controller

All layers required
Intra-layer configuration
Design Toolchain

Compiler Toolchain
32-point DCT Microarchitecture

- 2 Custom FUs, 2 RAM, 1 ROM, 3 ACU, 2 I/O ports
- Designer responsible for creating custom units manually

Retargetable Customization

- Prototype 2 technologies in OptimoDE
  - Automated ISA customization
  - Retargetable customization to an "application-area"
- Customizing for 1 application
  - Programmability → Nominally programmable
  - Critical problem – Cannot sustain performance across similar applications
  - How well does a custom ISA generalize
    - 5 encryption algorithms, create custom design for each
    - Average loss >80% versus native [MICRO, 2003]
  - Proactive generalization creates a retargetable design
Creating Custom Instructions

- Candidate discovery
  - Identify customization opportunities
- Examine program DFG
- Partition DFG at:
  - Memory operations
  - Unprofitable edges
- Enumerate candidate subgraphs within each partition

Grouping and Selection

- Group candidate subgraphs with same structure
- Estimate performance and cost for each group
- Greedily select groups to implement in hardware subject to budget
- 1 CFU created per group
Proactively Generalize Groups

- Cost-effectively extend group functionality to enable reuse
- Wildcard – multiple functionality at nodes
- Subsumed – configurable interconnect to bypass nodes

Native Speedups

- ARM 926EJ (200 MHz)
- OptimoDE (333 MHz, 5 Issue: 3 ALU, 1 Mem, 1 Brn)
- OptimoDE + 1 CFU (333 MHz)
Importance of Generalization

Case Study - Md5
OptimoDE Design for this Point

Die Area Breakdown

OptimoDE = 5.5 mm² in 0.13 µ
ARM 926EJ = 5.0 mm² in 0.13 µ
Conclusions

• OptimoDE
  – Configurable VLIW-style data engine architecture
  – Automated tools for implementing embedded signal and data processing solutions

• Automatic retargetable customization
  – Customized design combined with cost-effective generalization
  – Performance programmability - Performance stability across a family of similar applications

For More Information

• CCCP group website
  – cccp.eecs.umich.edu

• ARM OptimoDE information
  – www.arm.com/products/CPUs/families/OptimoDE.html
Designing for a Domain

![Graph showing the fraction of native speedup attained for different applications designed for a domain. The graph compares 3DES, AES, Blowfish, RC4, SHA, MD5, and the mean speedup for various applications.]